IN THE CLAIMS

Please amend the claims as follows:

1.   (currently amended)   A method ~~for procedural abstraction of instruction sequences in a computer program comprising the steps of~~ comprising:

-creating a control flow graph of [[the]] <u>a computer</u> program <u>having instruction sequences</u>, said <u>control</u> flow graph including basic blocks of instructions [[(504)]],

-traversing through the basic blocks in order to detect multiple occurrences of a same instruction sequence [[(506)]],

-creating a function including [[the]] <u>a</u> longest sequence <u>of last instruction sequences</u> common to at least two ~~instruction sequences~~ <u>basic blocks, said longest sequence</u> from a plurality of sequences <u>of last instruction sequences common to said at least two basic blocks and</u> having a common instruction sequence of equal or shorter length compared to said longest sequence [[(512)]], <u>said longest sequence including the equal or shorter length sequences of said plurality of sequences,</u> and

-replacing the original occurrences of said instruction sequences in said plurality of sequences with a reference to a proper position in said created function [[(514)]].

2. (currently amended)   A method of claim I, wherein the blocks are traversed in ~~opposite~~ <u>a</u> direction <u>opposite</u> to ~~the block~~ execution <u>of said blocks</u>.

3. (original) A method of claim 1, wherein said proper position is the position from which onward the sequence in the function matches with the original occurrence of the replaced instruction sequence.

4. (original) A method of claim 1, wherein said reference is substantially a function call or a branch instruction.

5. (currently amended) A method of claim 1, wherein said created function contains substantially the at least two basic [[block]] blocks whereto said longest sequence belongs.

6. (canceled)

7. (original) A method of claim 1, wherein after creating the flow graph said basic blocks are divided into a plurality of block sets, said blocks in different sets comprising no common instruction sequences.

8. (currently amended) A computer program product comprising code means to execute the method steps of claim 1 stored on a readable medium for execution by a processing unit so as to carry out:

- creating a control flow graph of a computer program having instruction sequences, said control flow graph including basic blocks of instructions,

-traversing through the basic blocks in order to detect multiple occurrences of a same instruction sequence,

-creating a function including a longest sequence of last instruction sequences common to at least two basic blocks, said longest sequence from a plurality of

sequences of last instruction sequences common to said at least two basic blocks and having a common instruction sequence of equal or shorter length compared to said longest sequence, said longest sequence including the equal or shorter length sequences of said plurality of sequences, and

-replacing the original occurrences of said instruction sequences in said plurality of sequences with a reference to a proper position in said created function.

9. (currently amended) A carrier medium carrying [[the]] a computer executable program of claim 8 for carrying out the method of claim 1.

10. (currently amended) An electronic device for abstracting instruction sequences in a computer program, said device comprising:
- a processing means (606) and unit,
- a memory means (602) for processing and storing instructions and data, and
- a data transfer means (608) module for accessing data,
said device arranged to create a control flow graph of [[the]] a computer program having instruction sequences, said control flow graph including basic blocks of instructions, said device further arranged to traverse through the basic blocks in order to detect multiple occurrences of a same instruction sequence, to create a function including [[the]] a longest sequence of last instruction sequences common to at least two instruction sequences basic blocks, said longest sequence from a plurality of sequences having a common instruction sequence of equal or shorter length compared to said longest sequence, said longest sequence including the equal or shorter length sequences of said plurality of sequences and to replace the original occurrences of said instruction sequences in said plurality of sequences with a reference to a proper position in said created function.

4

11.(new) The electronic device of claim 10 further arranged so that the blocks are traversed in a direction opposite to execution of said blocks.

12.(new) The electronic device of claim 10, arranged so that said proper position is the position from which onward the sequence in the function matches the original occurrence of the replaced instruction sequence.

13.(new) The electronic device of claim 10, arranged so that said reference is substantially a function call or a branch instruction.

14.(new) The electronic device of claim 10, arranged so that said created function contains substantially the at least two basic blocks whereto said longest sequence belongs.

15.(new) The electronic device of claim 10, arranged so that after creating the flow graphs said basic blocks are divided into a plurality of block sets, said blocks in different sets comprising no common instruction sequences.

16.(new) An electronic device comprising:
- means for processing,
- means for storing instructions and data, and
- means for accessing data,
said device arranged to create a control flow graph of a computer program having instruction sequences, said control flow graph including basic blocks of instructions, said device further arranged to traverse through the basic blocks in order to detect multiple occurrences of a same instruction sequence, to create a function including a longest sequence of last instruction sequences common to at least two basic blocks, said longest sequence from a plurality of sequences having a common instruction sequence of equal or shorter length compared to said longest sequence,

5

said longest sequence including the equal or shorter length sequences of said plurality of sequences and to replace the original occurrences of said instruction sequences in said plurality of sequences with a reference to a proper position in said created function.

17.(new) The electronic device of claim 16, wherein the blocks are traversed in a direction opposite to execution of said blocks.